

Absolvování individuální odborné praxe

Individual Professional Practise in the Company

Souhlasím se zveřejněním této bakalářské práce dle požadavků čl. 26, odst. 9 *Studijního a zkušebního řádu pro studium v bakalářských programech VŠB-TU Ostrava*.

V Ostravě 7. května 2010

.....

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. května 2010

.....

Rád bych na tomto místě poděkoval panu Ing. Bronislavu Kantorovi za vlídné přijetí do pracovního kolektivu a především za trpělivost a užitečné rady při vývoji programu.

Abstrakt

Cílem této bakalářské práce je popis mé práce, kterou jsem vykonal ve firmě Autel, a.s., se sídlem Oldřichovice 790, 739 61 Třinec. Ze začátku se věnuji zadání, které mi bylo zadáno. V průběhu práce se dostávám k jejímu řešení. Většina částí této práce popisuje celkový vývoj projektu. Projekt je vytvořen v Eclipse v programovacím jazyku Java. Na závěr této práce shrnuji nabyté zkušenosti z této práce.

Klíčová slova: JAVA, XML, praxe, Autel, eMonitor

Abstract

The main aim of this bachelor thesis is description of my job for company which name is Autel, a.s.. The residence of this company is Oldřichovice 790, 739 61 Třinec. At the beginning of this thesis I wrote about submissions which were given to me. In the further parts I wrote about solutions of submissions. Major part of this thesis is focused on global development of project. The project is created in computer program which name is Eclipse. The programming language which I was working with is Java. My job experiences are summarized in fine of this thesis.

Keywords: JAVA, XML, work experience, Autel, eMonitor

Seznam použitých zkratk a symbolů

a.s.	– akciová společnost
GUI	– Graphical user interface
XML	– Extensible Markup Language
IT	– Information technology
OOP	– Object-oriented programming
SQL	– Structured Query Language
MS	– Microsoft

Obsah

1	Úvod	3
2	PROFIL SPOLEČNOSTI A PRACOVNÍ ZAŘAZENÍ	4
2.1	Profil Společnosti	4
2.2	Pracovní zařazení	4
2.3	eMonitor	4
3	ZADANÉ ÚKOLY A ŘEŠENÍ	6
3.1	Příprava na vývoj projektu, seznámení se s nástroji, databází a eMonitoru	6
3.2	Rozvoj systému pro monitoring energií	6
4	ZÍSKANÉ, SCHÁZEJÍCÍ ZNALOSTI A DOVEDNOSTI	11
4.1	ZÍSKANÉ ZNALOSTI A DOVEDNOSTI	11
4.2	SCHÁZEJÍCÍ ZNALOSTI A DOVEDNOSTI	11
5	ZÁVĚR	12
6	Reference	13

Seznam obrázků

1	GUI Aplikace	14
2	GUI Aplikace	14

1 Úvod

Koncem 4. semestru bakalářského studia jsme si měli vybrat témata bakalářských prací. Já jsem si zvolil absolvování bakalářské práce formou praxe ve vybrané firmě. Začátkem září jsem byl pozván na pohovor do firmy Autel, a.s., která sídlí v mém rodném městě - Třinci. Zde mi bylo nabídnuto, jestli bych se rád podílel na vývoji a rozšíření již stávajícího systému. Celá aplikace je psána v jazyce Java, který je mi velice blízký, tudíž jsem neváhal a s radostí přijal tuto nabídku. Začátky mé praxe byly ve zkoumání zdrojových kódů aplikace eMonitor. Hlavní náplní mé praxe bylo rozšíření tohoto systému o novou funkcionalitu.

2 PROFIL SPOLEČNOSTI A PRACOVNÍ ZAŘAZENÍ

2.1 Profil Společnosti

AUTEL je inženýrsko-dodavatelská společnost zajišťující spolehlivé a pružné služby a dodávky nejmodernějších technologií na klíč pro průmyslové celky v těchto oblastech:

- VN, NN technologický silnoprúd a regulované pohony
- prostředky MaR, tzv. „polní instrumentace“
- řídicí a vizualizační systémy základní úrovně
- rozvrhování a řízení výroby (MES)

AUTEL během své existence byl a je členem týmu spolu se společnostmi reprezentujícími špičku světové technologie a vyvíjejícími stále nová řešení. Její zákazníci jsou rozmístěni po celém světě.

2.2 Pracovní zařazení

Na začátku měsíce září jsem byl pozván na vstupní pohovor, kde byli přítomni vedoucí oddělení IT Ing. Ivan Zboran a můj budoucí konzultant, pan Ing. Bronislav Kantor. Zde mi bylo nabídnuto, jestli bych se rád podílel na vývoji a rozšíření již stávajícího systému. Celá aplikace je psána v jazyce Java, který je mi velice blízký, tudíž jsem neváhal a s radostí přijal tuto nabídku. Již z pohovoru bylo jasné, že se budu podílet na rozšíření stávajícího systému. Mé místo ve firmě nebylo nějak zařazeno, používal jsem jak mé analytické schopnosti, tak i programátorské, ale kdybych měl mé místo pojmenovat, byl by to programátor v jazyce Java s analytickými prvky. Na mou práci dohlížel Ing. Kantor, s kterým jsem konzultoval své analýzy, problémy a vždy mě nasměroval tou správnou cestou, kudy by se měly mé myšlenky ubírat.

2.3 eMonitor

Hlavní náplní mé praxe, byl vývoj projektu: Rozvoj systému pro monitoring energií. Měl jsem rozšiřovat systém eMonitor. Tento systém se používá pro monitoring a bilan-cování medií. Je nasazován ve velkých průmyslových objektech, kde se například sleduje spotřeba tepla, vody, elektrické energie. eMonitor můžeme dále rozdělit podle funkce:

- Sběr a archivace dat.
- Monitoring, trendy, diagnostika.
- Dispečink, analýza odběru.
- Technické a finanční bilance.
- Metrologie a konfigurace.

- Reporty a exporty.

Měl jsem za úkol se věnovat monitoringu, trendům a diagnostice. Z programátorského hlediska jsem se zabýval prezentační vrstvou, kterou již vidí klient. Monitoring lze dále rozdělit na tyto části:

- Grafické průběhy měření.
- Tabulková data.
- Online sledování.
- Diagnostika měření.

3 ZADANÉ ÚKOLY A ŘEŠENÍ

3.1 Příprava na vývoj projektu, seznámení se s nástroji, databází a eMonitoru

V začátcích mé praxe mi byl vysvětlen systém eMonitor. Než jsem se pustil do samotného programování, byly mi přiděleny dva úkoly, seznámit se s vývojovým prostředím Eclipse a databází, kterou eMonitor využívá.

3.1.1 Eclipse Galileo 3.5

Eclipse je open source vývojová platforma, většina programátorů však tuto platformu zná jako vývojové prostředí určené pro programování v jazyce Java. Až doteď mé projekty byly vyvíjeny ve vývojovém prostředí Netbeans, které je také open source, a lze jej využít pro programování v jazyce Java. Mým prvním úkolem bylo stáhnout a nainstalovat Eclipse a seznámit se s tímto vývojovým prostředím do takové míry, abych se v něm cítil jistý.

3.1.2 Studování databáze a zdrojových kódů eMonitoru

V prvním měsíci mé praxe jsem se měl seznámit s aplikací eMonitor. Během tohoto měsíce jsem zkoumal zdrojové kódy a snažil se pochopit jednotlivé souvislosti. Při studování kódů mi docházelo, jak je to objektové programování krásné, jak si jednotlivé objekty mezi sebou vyměňují informace, jak čisté toto paradigma ve skutečnosti je. Žasl jsem údivem, jak kompaktně celá aplikace vypadá, jak všechno do sebe zapadá. Při studiu jsem měl i já tu možnost přidat pár řádku zdrojových kódů. Mé nadšení z nové práce či zkušenosti nadále rostly, když mi byla dána možnost nahlédnout do srdce celé aplikace a tím jest databáze. Takhle rozsáhlou databázi jsem viděl poprvé v mém životě. Byl to jeden velký koncert tabulek řízenými procedurami. Já jsem nechápal, když nám páni profesori tloukli do hlavy, že je vhodné řešit část logiky již na straně databáze. Zde jsem pochopil, že měli pravdu. Jelikož přenášet takové množství informací na klienta a následně to zpracovat by nebylo optimální a celý výkon aplikace by se rapidně snížil. I zde v databázi mi byla dána možnost napsat své myšlenky prezentované zdrojovými kódy. Měsíc s měsícem se sešel a já jsem dostal za úkol rozšířit tento systém o novou funkcionalitu.

3.2 Rozvoj systému pro monitoring energií

Tomuto projektu jsem se již věnoval celou dobu mé praxe. Ing. Kantor mě seznámil se svou myšlenkou, jak by si představoval program, který mi byl zadán. Jeho představa byla taková, že na základě definice, která bude uložena v xml souboru, se vytvoří celá aplikace. Tato aplikace bude umět načítat data ze serveru a také je korektně zobrazovat uživateli. Celý vývoj aplikace jsme rozdělili do několika etap, kde každá etapa byla řádně otestována, než se přešlo k další části vývoje aplikace.

3.2.1 Rozdělení vývoje do několika etap

- Analýza zadání a stanovení technologií.
- Vytvoření základních rozhraní pro popis definice.
- Vytvoření parseru a pokus zparsovat xml soubor.
- Vytvoření GUI komponent, datových objektů a loaderu.
- Práce s databází.
- Uprava kodu na filosofii klient-server.
- Refaktoring kódů a testování aplikace.

3.2.2 Analýza zadání a stanovení technologií

Při vývoji složitějšího programu je vhodné se zabývat důkladnou analýzou celého projektu. Již v této etapě můžeme objevit skryté problémy, které bychom museli později složitě řešit. Také jsme se zde zabývali tím, jaké technologie budeme využívat.

3.2.2.1 Grafické uživatelské rozhraní Součástí moderní Informační systému je uživatelsky přívětivé grafické rozhraní. Kladli jsme tedy důraz na grafickou stránku této aplikace. Na začátku celého projektu jsme navrhli, jak by měl program vypadat. Bylo nejjednodušší vzít prázdný papír, tužku a navrhnout GUI celé aplikace. Již z návrhu bylo zřejmé, že se pustím do velice zajímavého projektu, kde se budu zabývat i grafy, které jsem ještě v Javě neprogramoval. Jelikož firma Autel již nějakou dobu vytváří IS, byly mi poskytnuty jejich zdrojové kódy na tvorbu GUI.

3.2.2.2 XML Hlavní myšlenkou projektu je, že na základě definice uložené na serveru se sestojí celá aplikace. Po domluvě a zvážení pro a proti byl pro popis definice zvolen značkový jazyk XML.

3.2.3 Vytvoření základních rozhraní pro popis definice

V této části vývoje bylo zapotřebí vytvořit rozhraní, které bude umět uchovávat informace načtené z xml souboru. Jelikož xml je vlastně stromová struktura, je třeba také vytvořit sadu rozhraní, která budou představovat jednotlivé uzly xml souboru. Základem celé hierarchie rozhraní je `IPanelDefinition`, které si uchovává nastavení GUI, zejména počet řádků a počet sloupců. Dalším uzlem je rozhraní `IPanelComponentDefinition`, toto rozhraní představuje konkrétní komponentu, která se bude v aplikaci zobrazovat. Uchovává si pozice X, Y a také výšku a šířku komponenty. Dalším důležitým parametrem je název třídy, s kterou je tato komponenta svázána. Tato třída neboli objekt se skládá ze dvou důležitých informací. Za prve uchovává informaci o vizuální stránce, jak má daná komponenta vypadat. Za druhé uchovává informaci o loaderu, který se stará o načítání dat

ze serveru. Rozhraní `IpanelComponentDefinition`, jak jsem psal výše, si drží informace o GUI a Loaderu, tyto informace jsou reprezentovány rozhraními `IpanelDisplayDefinition` a `IpanelDataDefinition`.

3.2.4 Vytvoření parseru a pokus zparsovat xml soubor

Rozhraní již bylo navrženo a připraveno k použití. Bylo zapotřebí vytvořit třídu, která se postará o zpracování xml souborů a vytvoří definici rozhraní. V této chvíli nám pomohla analýza a naše volba použít pro definici xml. Java jako programovací jazyk plně podporuje práci s těmito soubory. Zapátral jsem v dokumentaci Javy a nastudoval třídu `DocumentBuilder`¹, která je dobrým pomocníkem pro zpracování xml. Po důkladném testování, opravování chyb byl Parser připraven k použití.

3.2.5 Vytvoření GUI komponent, datových objektů a loaderů

3.2.5.1 GUI Když bylo vytvořeno rozhraní i parser, začali jsme se zabývat vizuální stránkou aplikace. Jelikož se zabývám rozšíření stávajícího systému, bylo mi poskytnuto pár knihoven, které zpřijemňují tvorbou GUI. Při tomto studiu jsem se zaměřil na komponentu, která se stará o vizualizaci grafu, neboť graf v tomto projektu byl důležitou součástí. Důležitou částí bylo také vymyslet návrh jednotlivých tříd, aby nedocházelo ke zbytečné duplicitě kódů. V tomto ohledu je OOP silné paradigma. Po konzultaci jsme dospěli k závěru, že základ vizuální komponenty bude abstraktní třída, která na základě definice, kterou jí předáme, sestojí celé GUI. Od této třídy se již budou odvozovat konkrétní vizuální komponenty, které se budou lišit tím, jak budou zobrazovat data. Na sestrojení vzhledu komponenty mi bylo doporučeno nastudovat sadu tříd `JGOODIS`², které je na tvorbu dynamického vzhledu ideální.

3.2.5.2 Datové objekty Důležitou součástí aplikace jsou datové objekty, které představují konkrétní data získané ze serveru. Můžeme si to přiblížit na příkladě. Velice zjednodušeně, mějme továrnu - ta má různé spotřeby energie, jako je teplo, voda, plyn atd. Tato spotřeba je snímána snímači, která je dále zaslána na server. Naše aplikace zašle na tento server požadavek a dostává jako odpověď tento datový objekt. Tento datový objekt nese informaci, která byla nasnímana snímačem, se kterou pak nadále můžeme pracovat. Datový objekt neustále putuje mezi naší aplikací a serverem a získává cenné informace, které pak zobrazujeme ve vizuální komponentě.

3.2.5.3 Loadery - Načítání dat Vizuální komponenty a datové objekty byly připraveny. Přesunuli jsme se ve vývoji k tvorbě loaderu. Zjednodušeně řečeno, jsou to objekty, které mají za úkol sestojit dotaz, který vyšlou do databáze a vrátí vizuální komponentě data. Tato část pro mě byla velice zajímavá, neboť se zde pracovalo s vlákny v Javě, se kterými jsem se ve svých projektech moc nesetkal. Ze začátku loadery nepracovaly s databází,

¹Více o třídě `DocumentBuilder` zde <http://java.sun.com/j2se/1.5.0/docs/api/javax/xml/parsers/DocumentBuilder.html>

²Více o tvorbě v `JGOODIS` zde <http://www.jgoodies.com/>

ale jen generovaly náhodná data, která předávala vizuální komponentám. Zatím pro testování aplikace to stačilo. Testoval jsem především synchronizaci vláken a pozoroval, jestli náhodou nedojde k nějakému uváznutí v rámci aplikace. Po důkladném testování se loadery upravily tak, aby sahaly přímo do databáze.

3.2.6 Práce s databází

Databáze je nedílnou součástí každého informačního systému a ani tady to nebylo výjimkou. Na chvíli jsem Javu opustil a věnoval se databázi. Pro vývoj aplikace se využívala testovací databáze, která mi byla poskytnuta první den práce. Taky mi byl přidělen server MS SQL Server 2005 Express, kde databáze běžela. Ze začátku jsem studoval testovací databázi, která má veliký počet tabulek. Postupem času jsem přidával i já tabulky. Dále bylo zapotřebí vytvořit několik procedur, které řešily logiku mezi těmito tabulkami. Procedury na straně databáze jsou velice výhodné, když máme desítky tabulek. Nemusíme řešit logiku dotazování na klientské části a tím zatěžovat režii celé aplikace, ale předáme část logiky na stranu databáze. Když byla databáze upravená, znovu jsem se vrátil k Javě. Bylo zapotřebí upravit loadery tak, aby načítaly data z databáze. Po této úpravě byla aplikace plně funkční. Chvíli jsem tuto aplikaci testoval a pak ji prezentoval panu Kantorovi.

3.2.7 Úprava kódu na klient-server aplikaci

Jelikož jsem rozšiřoval eMonitor, bylo zapotřebí aplikaci přizpůsobit tak, aby mohla být bez problému integrována. Byly mi poskytnuty další zdrojové kódy, které bylo zapotřebí nastudovat k pochopení a použití pro vyvíjenou aplikaci. eMonitor se skládá s mnoha aplikací. Tyto aplikace zasílají své požadavky/requesty na server, který vrací odpověď/response. Prostředník mezi požadavkem a odpovědí je dispatcher. Tento dispatcher se postará o zpracování dat z databáze a vrátí je zpátky klientovi. Tento úkol jsem si rozdělil na části:

- Vytvoření requestu.
- Vytvoření dispatcheru.
- Úprava loaderů.

Po těchto úpravách byla naplněna myšlenka, která na základě definice uložené na serveru vrátí plně funkční aplikaci. Poslední část vývoje programu bylo testování a refaktoring.

3.2.8 Refaktoring kódů a testování aplikace

V závěru mého působení ve firmě spočívala má práce v refaktoringu kódu. Refaktoring v této souvislosti znamenal:

- Odstranění nadbytečných tříd, balíčků, souborů.

- Přejmenovávání jednotlivých tříd za účelem přehlednosti.
- Komentování důležitých částí kódu.
- Přemísťování tříd mezi balíčky.
- Poupravení názvů metod, proměnných.

Na závěr jsem aplikaci důkladně testoval a hledal případné chyby, které se mohly vyskytnout. Po této části byla aplikace připravena k odevzdání.

4 ZÍSKANÉ, SCHÁZEJÍCÍ ZNALOSTI A DOVEDNOSTI

4.1 ZÍSKANÉ ZNALOSTI A DOVEDNOSTI

Tato praxe byla pro mě nesmírným přínosem. Rozšířil jsem si znalosti v programovacím jazyku Java, a to zejména tím, když jsem pracoval s vlákny a xml soubory. Dále mi přinesla tato praxe větší zkušenosti s návrhem aplikace. Naučil jsem se lépe využívat prvky OOP, především polymorfismu, dědičnosti a práce s rozhraním. Naučil jsem se také orientovat v cizích kódech, které byly využity k tvorbě aplikace. V neposlední řadě беру jako velkou zkušenost a přínos této praxe v práci s databází.

4.2 SCHÁZEJÍCÍ ZNALOSTI A DOVEDNOSTI

Před nástupem do praxe jsem měl teoretické znalosti, ale žádné nebyly prověřeny praxí. V začátcích mého působení ve firmě jsem především měl za úkol studovat a upravovat cizí kódy. Měl jsem problémy se v těchto kódech orientovat. Také mi scházela praxe v pokročilejších programovacích technikách v Javě, jako je například práce s reflexí, s vlákny a s dokumenty xml.

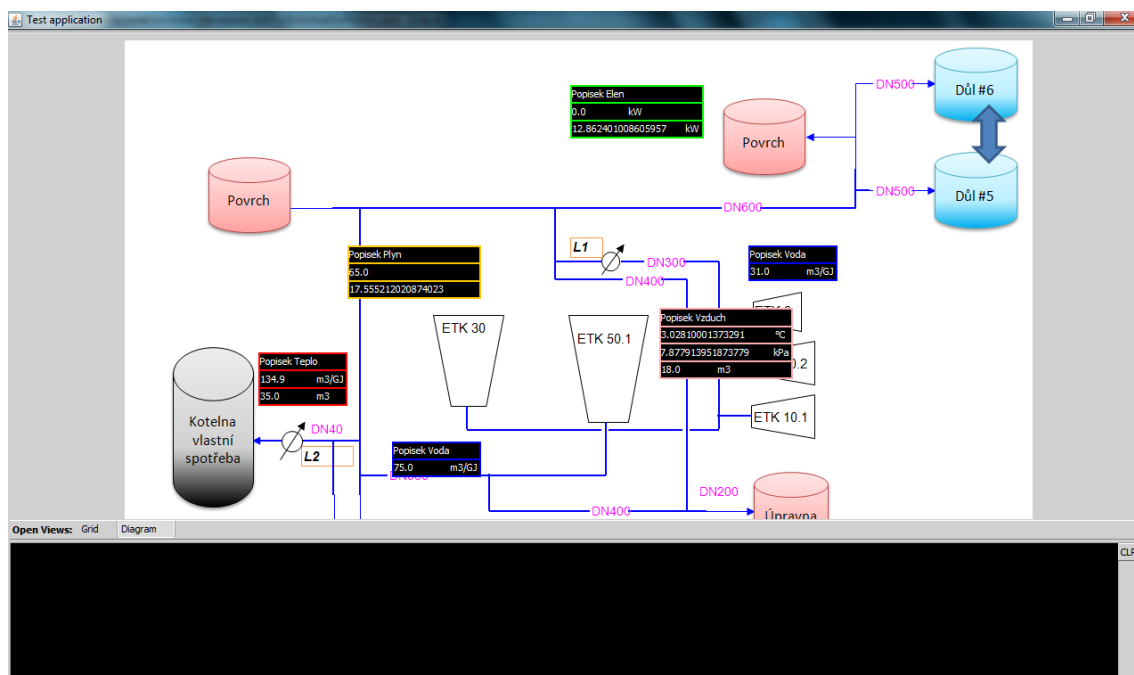
5 ZÁVĚR

Na závěr bych rád dodal, že podílet se na vývoji programu bylo nesmírně zajímavé. Pod dohledem zkušeného vývojáře jsem mohl sledovat, jak z náčrtku na papíře vzniká ucelená aplikace, která dokáže načítat data ze serveru a následně je zobrazovat uživateli. Jsem velice rád, že mi byla dána tato možnost a věřím, že tyto zkušenosti, které jsem získal, uplatním v budoucím povolání. Ještě jednou bych rád poděkoval panu Ing. Bronislavu Kantorovi, za trpělivost a cenné rady při vývoji programu.

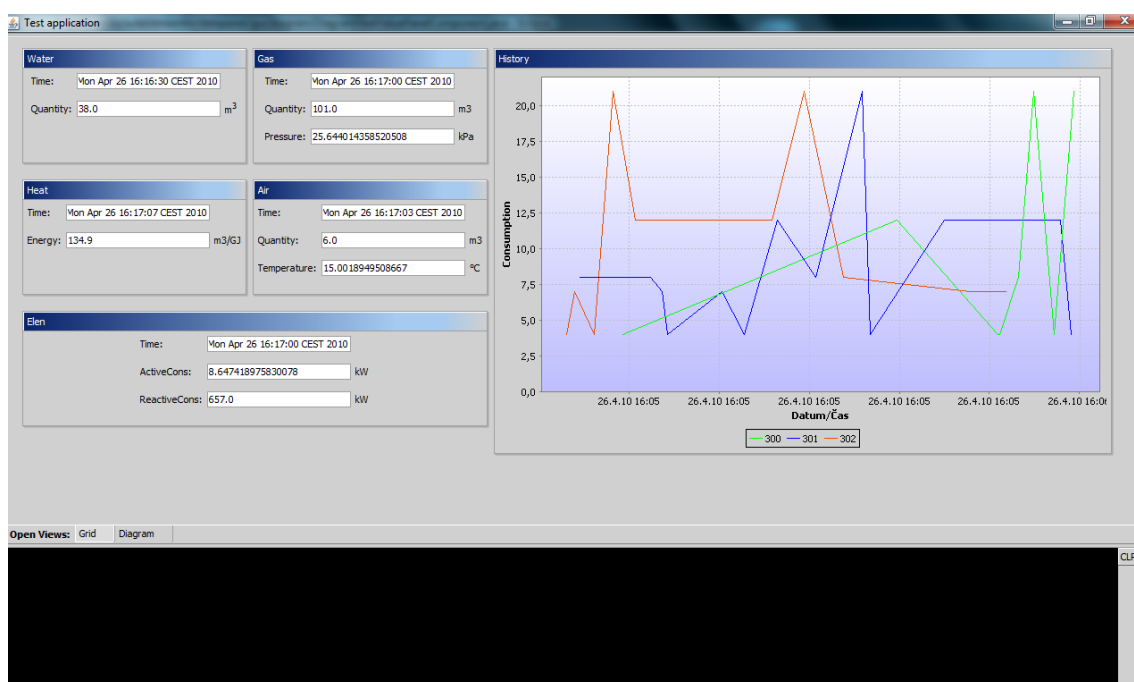
Marek Garbulinský

6 Reference

- [1] Autel, a.s. [online], <http://www.autel.cz/o-spolecnosti-profil-spolecnosti.aspx> [citováno 26. dubna 2010]
- [2] Autel, a.s. [online] <http://www.autel.cz/index.aspx> [citováno 26. dubna 2010],



Obrázek 1: GUI Aplikace



Obrázek 2: GUI Aplikace